Supplementing Introductory Experiences With Worked Examples

Michael Friend under direction from Austin Cory Bart

Problem

- Interest in computing is growing, leading to huge introductory computing class sizes [NAS].
- Further, many assignments in these courses can be challenging for students, even when they only exercise a few concepts [RAINFAILL].
- Worked Examples have shown promise as a scaffold to help students complete programming assignments.
- However, there have been limited classroom studies to evaluate the effectiveness of Worked Examples.

Hypothesis

Adding worked examples related to difficult problems will improve understanding of the problems and their ability to program

Research Questions

- Do WEs improve performance?
- Do students take advantage of worked examples?
- Do students find WEs helpful?

Prior Work

- Prior work suggests providing WEs with clear subgoal labels help students deconstruct problems. [2]
- Subgoal labeled instructional text paired with subgoal labeled examples can improve performance [1]

Educational Theories

- Worked Examples [5][6]
- Example-Problem Pairs [4]
- Subgoal Learning [2]

Context

- Introduction to Programming in Python
- Non-CS majors from mostly Engineering and Sciences
- Students complete 188 programming assignments in and online programming environment
- Students had infinite tries over two weeks for each assignment
- Keystroke level edits and environmental interactions are logged

	Prior Experience	No Prior Experience	Total
F17	240	41	281
S18	197	43	240
Total	437	84	

Methodology

Manual inspection of prior semester used to determine "hard" problems
- "Hard" problems were those that took most students more than 20 edits to complete

→ Worked examples developed for the 8 hardest problems to help students

→ Links to relevant worked example provide on targeted problems

→ Performance and usage data collected for problems with worked examples

Sample WE Text

Worked Example: Pet Count

0) Read Problem

Write a function called count_pets that consumes a list of strings of pet types owned and returns the number of each type of pet as a dictionary. Use the Dictionary Counting pattern. Call your function with the following list to test it.

["Dog", "cat", "Cat", "Snake", "mouse", "snake", "dog", "dog"]

Notice the list is written somewhat sloppy. The capitalization is inconsistent. Your function needs to be able to fix this so that "Dog" and "dog" go into the same dictionary entry.

1) Interpret the Problem

This problem requires us to write a function that uses the Dictionary Counting pattern to create a dictionary that maps pet type to number of pets of that type. The problem also says the words in the list can have inconsistent capitalization and our function has to treat two of the same word as one key in the dictionary.

Data Collection

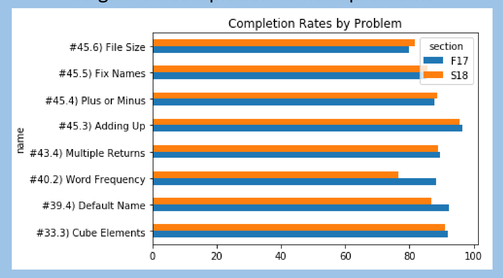
- Qualitative Data collected via a survey on student opinion and usage of Worked Examples
- Quantitative Data collected via exercise completion rates and student interaction with Blockpy and Worked Example
  - Worked Examples page was instrumented to log student interaction

Results

64% of students found the Worked examples helpful.

	Prior Experience	No Prior Experience	Total
WE Helpful	108	29	137
WE Not Helpful	66	12	78
Total	174	41	

The addition of Worked Examples provided little to no gain on completion of hard problems



Measured usage of WE varied widely by problem, with an average around 50

Problem	Percent Usage
Cube Elements	29.4
Default Name	54.7
Word Frequency	59.0
Adding Up	33.3
Plus or Minus	60.1
Fix Names	60.5
File Size	41.2

Students who used WEs actually completed the problem in more runs/time than those that didn't

Section	Used WE	Prior Experience	Average # Runs	Standard Deviation	Average Time on Task (sec.)	Standard Deviation
F17	False	False	19.5	23.3	574	478
		True	13.8	20.1	415	393
S18	False	False	7.3	7.8	311	191
		True	6.6	8.3	274	238
	True	False	19.1	18.0	727	476
		True	17.4	19.5	707	501

- Focusing on the values for S18 students with no prior experience, we see those that used the WE actually took over twice the time and runs as those who didn't.
- A Cohen's D test gives a large effect size for both measures (p=0.8, p=1.1)

Conclusions

The Worked Example strategy we had minimal effect on the completion rate of problems. However, many students used them and found them helpful.

Methods of selection "hard' problems may have been flawed. All problems had high completion rates (80-90%) before introducing WEs.

The correlation between WE use and increased number of runs may suggest our implementation had no quantitative benefit. It may also simply show that students are more likely to use the WE if they have been working on the problem for longer. More analysis of event log is required to support this.

Future work

- Further analysis of WE usage via event logs
- Interactive Worked Examples
- Subgoal only Worked Examples
- Alternative supplements material in large intro classes

References

[1] Margulieux, L. E.; Catrambone, R., Improving problem solving performance in computer-based learning environments through subgoal labels. In *Proceedings of the first ACM conference on Learning @ scale conference*, ACM: Atlanta, Georgia, USA, 2014; pp 149-150.

[2] Morrison, B. B.; Margulieux, L. E.; Guzdial, M., Subgoals, Context, and Worked Examples in Learning Computing Problem Solving. In *Proceedings of the eleventh annual International Conference on International Computing Education Research*, ACM: Omaha, Nebraska, USA, 2015; pp 21-29.

[3] National Academies of Sciences, E.; Medicine, *Assessing and Responding to the Growth of Computer Science Undergraduate Enrollments*. The National Academies Press: Washington, DC, 2018; p 252.

[4] Skudder, B.; Luxton-Reilly, A., Worked examples in computer science. In *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148*, Australian Computer Society, Inc.: Auckland, New Zealand, 2014; pp 59-64.

[5] Renkl, A. (2005). The Worked-Out Examples Principle in Multimedia Learning. In R. Mayer (Ed.), *The Cambridge Handbook of Multimedia Learning* (Cambridge Handbooks in Psychology, pp. 229-246). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511816819.016

[6] Sweller, J.; Cooper, G. A., The Use of Worked Examples as a Substitute for Problem Solving in Learning Algebra. *Cognition and Instruction* **1985,** *2* (1), 59-89.